



SUBJECT: Operating System

SUBJECT CODE: 24CAT-105

Case study

STUDENT NAME: Aryan Raj

UID: 24BCA10093

BRANCH: BCA

SECTION/GROUP: 24BCA (2A)

SUBMITTED TO: Gagandeep Sir

Multitasking Operating System

Introduction

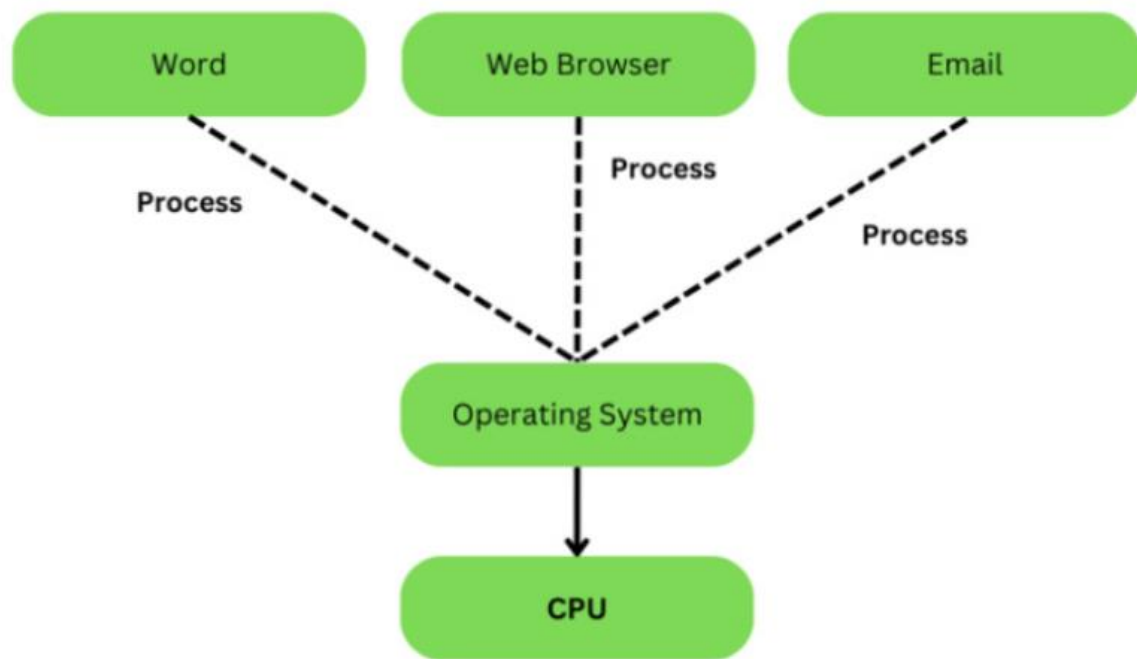
An OS that can manage numerous duties or procedures at once is known as a multitasking operating system. In simple terms, it enables the simultaneous operation of numerous programs or procedures while allocating a specific amount of memory and central processing time to each process.

Every task's distribution of resources is controlled by the operating system's kernel, which also ensures that no tasks conflict with one another. Additionally, it offers a way to move among duties rapidly, giving the impression that every process is active at once.

Contemporary systems for computing, which include personal computers, laptops, computer systems, and cellphones, frequently utilize multitasking operating systems. OS like Windows, Linux, macOS, Android, and iOS are a few prominent instances of multitasking systems.

Operating systems that support multiple tasks have a number of benefits over those that only support one task. They make multitasking simpler and increase productivity by enabling individuals to operate several programs simultaneously.

In this article, we will be discussing some use cases, examples, and components of a Multitasking Operating System.



Components of a Multitasking Operating System

The main components of a Multitasking Operating System are the following.

Process Creation – The `fork()` system call is used to create a new process, which results in two identical processes running concurrently.

Process ID – The `getpid()` function retrieves the process ID of the current process, while `getppid()` retrieves the process ID of the parent process.

Conditional Execution – The program uses if and else if conditions to differentiate between the parent and child processes and executes the corresponding code based on the process ID.

Use Cases of Multitasking Operating System

Concurrent Application Execution – One of the primary use cases of multitasking operating systems is the ability to execute multiple applications or processes concurrently. Users can run multiple programs simultaneously, switch between them, and perform tasks in parallel, enhancing productivity and user experience.

Resource Sharing and Allocation – Multitasking operating systems effectively allocate system resources such as memory, CPU time, and input/output devices among multiple processes. Each process is assigned a specific amount of resources to ensure fair and efficient utilization. This allows for better resource management and optimal system performance.

Faster Response Time – Multitasking operating systems enable quick context switching between processes, resulting in faster response times for users. Users can switch between applications seamlessly, perform tasks concurrently, and experience a smooth and responsive computing environment.

Enhanced System Stability – Multitasking operating systems implement safeguards to prevent a single program or process from affecting the entire system. If one program crashes or encounters an error, it does not impact other running processes, ensuring system stability and reliability.

Utilization of Multi-Core Processors – Modern hardware often includes multi-core processors and multitasking operating systems can effectively utilize these resources. By distributing tasks across multiple cores, the operating system

can improve overall system performance and take advantage of the parallel processing capabilities of the hardware.

Virtual Memory Management – Multitasking operating systems employ virtual memory techniques to allow each process to use more memory than is physically available. Virtual memory provides an illusion of abundant memory space by swapping pages of memory between physical memory and secondary storage, such as a hard drive. This enables efficient memory utilization and allows processes to access more memory than would otherwise be possible.

Advantages

There are several perks of using a Multitasking Operating System, which are as follows–

- Increased productivity – Individuals can work on multiple projects or programs at once using a multitasking operating system, which increases efficiency.**
- Better resource utilization – An OS that supports multiple tasks effectively divides up system components like RAM and central processing time among them.**
- Faster response time – Users experience faster response times thanks to the quick transition among programs made possible by multitasking operating systems.**
- Improved system stability – Linux systems that support multiple tasks have safeguards implemented to stop a single program from failing the whole system.**

- **Better utilization of hardware –** Contemporary hardware's multiple processing units or CPU cores can be used by multitasking operating systems to increase the efficiency of the system.
- **More efficient memory –** Memory that is virtual is used by multitasking systems to enable each of the processes to use greater storage space compared to what is actually accessible.

Disadvantages

There are several drawbacks to using a Multitasking Operating System, which are as follows –

- **Resource Contention –** There might be competition for computer assets like RAM and central processing time when multiple programs are active at once.
- **Security risks –** Operating multiple programs or procedures at once widens the machine's strike exterior. A flaw in a particular procedure may have been used to gain entry to flaws in other procedures, possibly placing the system as a whole at risk.
- **Complexity –** Operating systems that support multiple tasks tend to be more complicated than those that support just one endeavor, and that variety may render these individuals more challenging to operate and uphold.
- **Cost –** In order to function properly, multitasking operating systems need stronger hardware, and this

may prove more costly than the technological infrastructure needed for focused attention platforms.

- **Fragmentation** – This may be challenging to maintain every procedure that is active at once. As a result, documents as well as additional materials may become fragmented, and this gradually may worsen the system's efficiency.
- **Compatibility issues** – The variety of software that is accessible to individuals could be constrained by the incompatibility of some older programs with multitasking operating systems.

Conclusion

In summary, multitasking operating systems enable clients to operate numerous programs or procedures concurrently, enhancing system efficiency and increasing worker efficiency. Yet, employing an OS that can multitask possesses some disadvantages as well, including resource competition, security risks, complexity, and expense. To guarantee optimal efficiency and protection, appropriate administration of resources and comprehension is required. Multitasking platforms continue to be commonly utilized in contemporary computing, supporting laptops, desktop computers, computer systems, and smartphones regardless of these difficulties.